Linear Grouping Using Orthogonal Regression

Stefan Van Aelst *

Dept. of Applied Mathematics and Computer Science, Ghent University, Krijgslaan 281 S9, B-9000 Gent, Belgium.

Xiaogang (Steven) Wang

Dept. of Mathematics and Statistics, York University, Toronto, Ontario M3J 1P3, Canada.

Ruben H. Zamar

Dept. of Statistics, University of British Columbia, 333-6356 Agricultural Road, Vancouver, BC, V6T 1Z2, Canada.

Rong Zhu [1]

Dept. of Mathematics and Statistics, McMaster University, 1280 Main Street West, Hamilton, Ontario L8S 4K1, Canada

Abstract

A new method to detect different linear structures in a data set, called linear grouping algorithm (LGA), is proposed. LGA is useful for investigating potential linear patterns in datasets, that is, subsets that follow different linear relationships. LGA combines ideas from principal components, clustering methods and resampling algorithms. It can detect several different linear relations at once. Methods to determine the number of groups in the data are proposed. Diagnostic tools to investigate the results obtained from LGA are introduced. It is shown how LGA can be extended to detect groups characterized by lower dimensional hyperplanes as well. Some applications illustrate the usefulness of LGA in practice.

Key words: Linear grouping, orthogonal regression.

Key words: Linear grouping, orthogonal regression.
1991 MSC: 62H30, 68T10, 62P10

1 Introduction and motivation

Clustering, the method used to find groups in a dataset, has received enormous attention in the literature. In fact, clustering is an important tool for *unsupervised learning* where the dataset consists of n observations in d dimensions and we want to uncover properties of their joint distribution. Many clustering methods and algorithms have been proposed in various fields such as statistics (see e.g. Hartigan, 1975; Kaufman and Rousseeuw, 1990; Banfield and Raftery, 1993; Scott, 1992; Silverman, 1986; Murtagh, 1983), data mining (Ng and Han, 1994; Zhang, Ramakrishnan, and Livny, 1997; Bradley, Fayyad, and Reina, 1998; Murtagh, 2002), machine learning (Fisher, 1987), and pattern recognition (Duda and Hart, 1973; Fukunaga, 1990). Not all patterns causing different groups can be recognized by identifying sparse and crowded places. For example, in allometry studies considered in Section 4, some types of animal species form one linear relationship between their body weight and brain weight while some other types have another linear relationship. Standard clustering techniques are not able to find these linear patterns.

Clustering and linear grouping are often used in the context of unsupervised learning where there are not specified input and output variables. Indeed, in many situations calling for clustering or linear grouping it is not likely to have a naturally defined response variable available. On the other hand, many methods for linear grouping proposed in the literature - including Späth (1982, 1985); DeSarbo, and Cron (1988); DeSarbo, Oliver, and Rangaswamy (1989); Wedel and Kistemaker (1989); Kamgar-Parsi, Kamgar-Parsi, and Wechsler (1990); Gawrysiak, Okoniewski, and Rybiński (2000) - assume that an output variable is available.

To illustrate the problem we use the artificial example in Figure 1. This example consists of two equally sized groups each generated from a different linear structure. One group (marked $\circ$) follows the model $y = 10x + \varepsilon$ while the other group (marked $\triangle$) follows the model $y = -x + \varepsilon$. For both groups x and ε come from a Gaussian distribution with standard deviations respectively 5 and 15. The third dimension is a Gaussian variable, z, with mean zero and standard deviation 10. One method (Späth, 1982) to successfully separate the groups in Figure 1 is the following. First designate one of the three variables as response. Split the groups in two sets and find the ordinary least squares re-

* Corresponding author. Tel: +32-9-264-49-08; fax: +32-9-264-49-95.
Email addresses: Stefan.VanAelst@UGent.be (Stefan Van Aelst), stevenw@mathstat.yorku.ca (Xiaogang (Steven) Wang), ruben@stat.ubc.ca (Ruben H. Zamar), rzhu@math.mcmaster.ca (Rong Zhu).
[1] Part of this work was done while Rong Zhu was Postdoctoral Fellow, Pacific Institute of Mathematical Sciences, University of British Columbia, Vancouver, BC, V6T 1Z2, Canada.

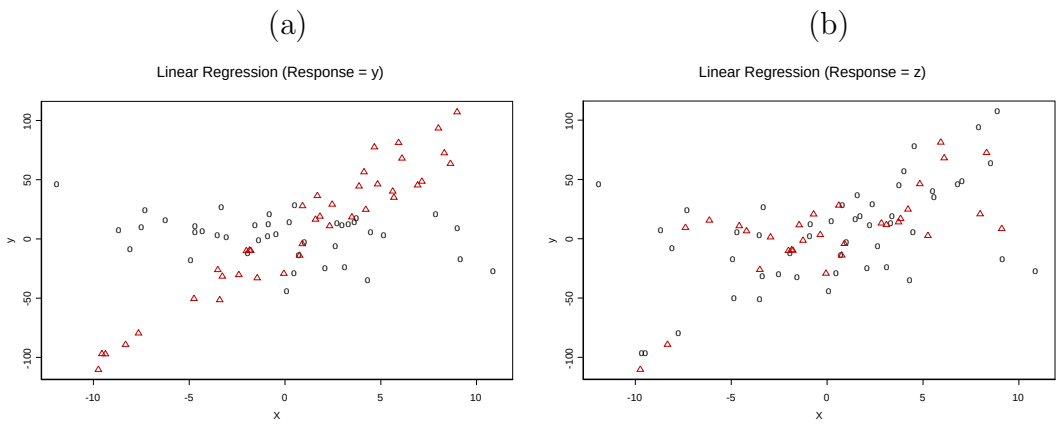

Fig. 1. Two groups detected with the algorithm based on least squares regression when (a) y is the response variable and (b) z is the response variable.

gression planes for each set. Now form two new groups by assigning each point to the closest plane and compute two new planes by applying least squares to each group. These steps are iterated to convergence.

If we apply the method based on least squares outlined above and specify y as the response variable, then we obtain the good result in Figure 1a. On the other hand, if we select z as response, then the method yields the groups in Figure 1b where the linear structures have been completely lost.

This shows that in general we would have to consider each variable as a possible output variable. To get around the problem of choosing the output variable, we instead search directly for different groups around $d-1$ dimensional hyperplanes. The hyperplane for each group is simply a translation of the subspace orthogonal to the smallest principal component of that group. More precisely, the hyperplane is defined by the equation $a^t x = b$ where a is the eigenvector associated with the smallest eigenvalue of the covariance matrix of the group and b equals the inner product between a and the group average. It is well known that this hyperplane (called the orthogonal regression hyperplane) is the closest in mean *orthogonal distance* to the points in the group. (See for example Johnson and Wichern, 1998). Moreover, this hyperplane is also the maximum likelihood solution for the linear error-in-variables model (see for example Fuller, 1987) which can also be formulated as a total least squares problem. Our approach makes it unnecessary to specify response variables but identifies functional relationships and therefore is better suited for the unsupervised learning setup. Note that although the groups follow different patterns, they may overlap (see Figure 1). Our technique can detect linear groups even in situations with heavily overlapping regions. Related methods applicable to 2 dimensional problems in this context are given by Murtagh and Raftery (1984); Phillips and Rosenfeld (1988).

Banfield and Raftery (1993) proposed a flexible clustering procedure (MCLUST) based on a mixture of normal distributions with covariance matrices of the same shape but different orientation and sizes (see also Woodruff and Reiners, 2004). This method is capable of identifying some linear groups but the algorithm still searches for clusters of points around a common center and therefore can miss some linear patterns. The (x, y) panel in Figure 2 reveals a clear linear grouping. Variable z is random noise. The grouping found by MCLUST - the top panel in Figure 3 - does not reflect the linear structures in the data. The groups were obtained using the Splus implementation of MCLUST with method=S^* and shape equal to $c(1, .01, .01)$ to favor linear structures. On the other hand the LGA solution - the bottom panel in Figure 3 - reveals well the two linear patterns from which the data were generated.

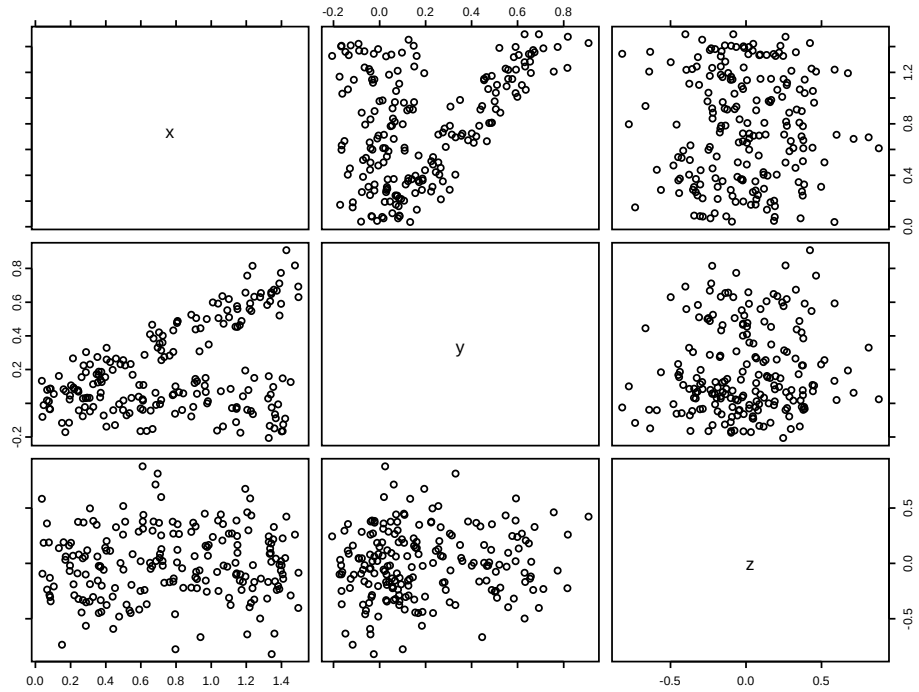

Fig. 2. Dataset with two groups generated according to different linear structures.

In practice the number of different linear groups is often unknown. Therefore, we propose procedures to determine the number of linear groups and compare them in several simulations. A related problem is to determine the strength of the linear grouping once it has been found. For solving this problem we extend the notion of silhouette values (Rousseeuw, 1987) to the linear grouping setting. Alternatively, Bayesian factors can be used to measure strength of group membership.

LGA is explained in Section 2. We discuss the problem of determining the number of groups in Section 3. In Section 4 we analyze some applications of

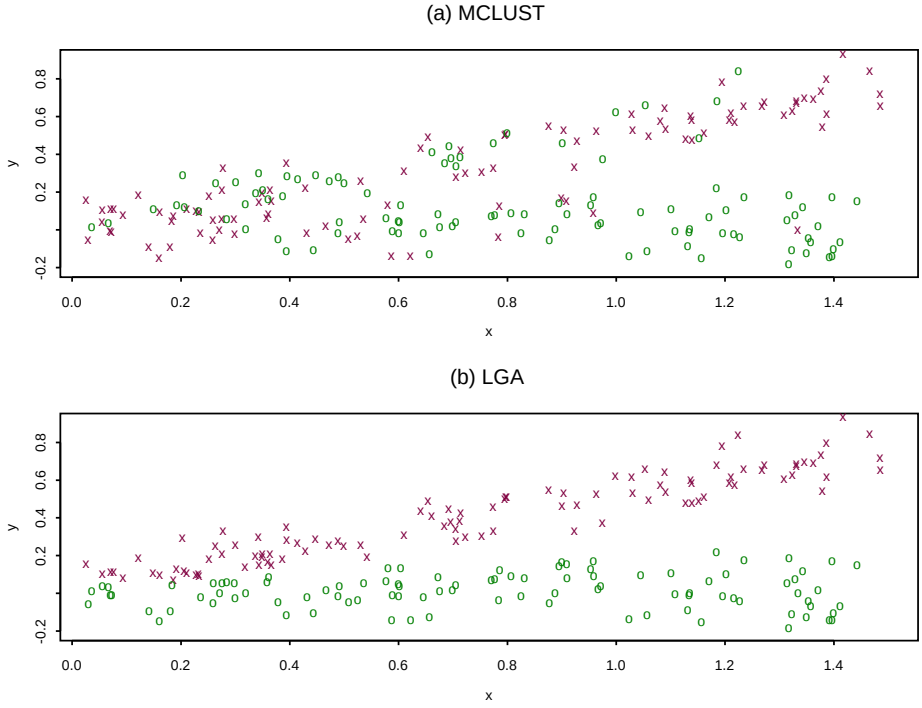

Fig. 3. Dataset with two groups generated according to different linear structures. Groups detected by (a) MCLUST and (b) LGA algorithms for $k = 2$ groups.

LGA while Section 5 introduces the diagnostic procedures to investigate the strength of the structure detected by LGA. In Section 6 we propose a procedure based on LGA to detect groups concentrated around lower dimensional hyperplanes. Section 7 concludes with a discussion.

2 Linear Grouping Algorithm (LGA)

We now present an algorithm capable of detecting different linear structures in a dataset. LGA uses orthogonal regression to identify the linear relationships and iterative optimization similar to K-means (Hartigan and Wong, 1979) to converge to a local minimum. Note that Pacheco and Valencia (2003) recently proposed several alternatives for K-means to solve the minimum sum-of-squares clustering problem. To increase the performance of the algorithm the iterative optimization is repeated a large number of times with different random starting values obtained by resampling. Finally, LGA reports the optimal solution in terms of the aggregated sum of squares of orthogonal residuals.

Consider a data set of size n in d dimensions. LGA is described in detail by the following steps:

1. Scaling of the variables. Each of the variables is divided by its standard deviation such that they have unit variance.

2. Generation of the starting values. Starting values are generated by randomly selecting k mutually exclusive subsets of d points (d-subsets). For each of these d-subsets we then compute the orthogonal regression hyperplane through these d points. This is a simple calculation exploiting the well-known connection between orthogonal regression and principal components. By using d-subsets to compute starting values we obtain intial solutions that are "closer" to the data which reduces the number of iterations in step 4.

3. Initialization of the groups. For each starting solution of k hyperplanes we compute the squared distances of all data points to these hyperplanes. We then assign each point to the closest hyperplane and recalculate the hyperplanes from this grouping.

4. Iterative refinement. The procedure in step 3 is repeated a small number of times for each of the starting values. Because the initial starting values are based on d-subsets, a few iterations (e.g. 10) usually suffices to determine which of the starting values will lead to the optimal solution (see also Rousseeuw and Van Driessen, 1999).

5. Resampling. Repeat Steps 2 to 4 a number of times (e.g. 100 times) and select the solution which has the lowest value of the objective function, given by the aggregated sum of the squared distances between the data points and their closest hyperplane. This solution can then even be iterated further (as in step 4) until no improvement is obtained anymore.

The iterative refinement in step 4 will converge to a good solution if the initial random start is already of high quality, that is when each of the initial hyperplanes is based on a majority of points from one of the groups. For random starts of low quality the iterative refinement will less fequently lead to a good solution. Hence, it is important to take enough random starts to have a high enough probability of having at least one start of higher quality. Therefore we proposed in Step 2 to compute random starts from d-subsets which is more likely to produce random starts of high quality than entire random selection of initial hyperplanes.

To get some guidance regarding the number of resamples in Step 5, we calculate the minimal number of starting values, m, needed to have 95% probability of obtaining at least one sample with d points from each group. The probability of getting such a sample is

$$p = \frac{\binom{n_1}{d}\binom{n_2}{d}\cdots\binom{n_k}{d}}{\binom{n}{kd}}$$

Table 1
Number of random starts for 95% probability of at least one good subset.

	$k=2$		$k=3$		$k=4$	
d	1:1	1:2	1:1:1	1:2:3	1:1:1:1	1:2:3:4
2	7(7)	9(10)	23(24)	42(43)	73(77)	201(206)
3	9(9)	13(13)	34(35)	82(83)	127(135)	580(586)
4	10(10)	17(17)	44(45)	145(145)	187(203)	1462(1431)
5	11(12)	52(51)	53(56)	244(239)	253(280)	3446(3207)

and therefore m satisfies the equation $1-(1-p)^m = 0.95$. That is

$$m = \frac{\log(0.05)}{\log(1-p)}.$$

Table 1 shows that the value of m depends on the number of groups, the relative sizes of the groups, and the dimension of the data, in that order. Fortunately, m doesn't depend much on the data size, n. Table 1 gives the values of m for $n = 300$ observations, $k = 2, 3, 4$ groups and $d = 2, 3, 4$ dimensions. We also considered two different situations regarding the degree of unbalance in the group sizes (e.g. 1:1:1:1 for four groups of equal size and 1:2:3 for three groups in the relation 1 to 2 and to 3). The number in parenthesis corresponds to the limiting case approximated by taking $n = 100,000$.

Our current implementation of LGA uses the values of m corresponding to equal group sizes as default. A higher number of random starts yields a higher chance of obtaining the optimal solution but is less time efficient. In our experience, when a strong grouping structure exists in the data, a moderate number of random starts (say between 10 and 50) suffices to find it.

3 Determining the number of groups

The number k of groups is a required input of LGA. In some applications k is suggested by background information such as gender, location, etc. However, in cases when such features are not available we need tools to determine the number of groups. Moreover, finding k may be a primary research interest.

Scatterplots provide visual information regarding the number of groups. As an illustration we consider a small dataset consisting of 31 measurements of the height and mass volume of trees (Ryan, Joiner, and Ryan, 1976). Figure 4 show the results of LGA for one, two and three groups. We see that volume increases with height, but the measurements become scattered for taller trees. Clearly, one group doesn't suffice for these data and three is too many. On

the other hand, the picture with two groups is visually appealing and has biological interpretation. Namely, the first group (labeled 2 in the picture) corresponds to older trees which tend to have larger girth, so their volume increases faster with height. The second group corresponds to younger, thinner trees. Unfortunately, scatterplots are mainly helpful in low (2 or 3) dimensions.

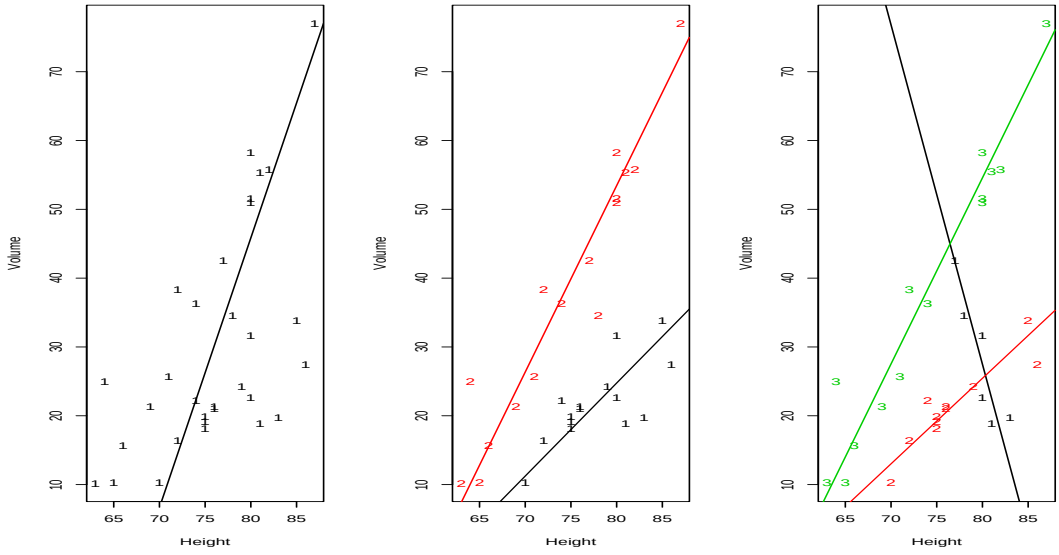

Fig. 4. The height and volume of young and old trees.

Even then, for heavily overlapping regions our eyes may fail to distinguish some linear patterns.

To determine the optimal number of groups we propose several criteria similar to methods that are available for clustering. Recently, Tibshirani, Walther, and Hastie (2001) proposed the GAP statistic as a very flexible method to estimate the optimal number of clusters in a data set. The GAP statistic compares the pooled within-cluster sum of squares around the cluster means with its expectation under a null reference distribution. To detect linear groups, LGA uses the orthogonal distance between a point and its associated hyperplane to measure how far the point lies from this hyperplane. Hence, the GAP statistic can easily be adapted for estimating the number of linear groups by replacing the pooled within-cluster sum of squares with the aggregated sum of the squared orthogonal distances. To generate data from the reference distribution the variables are generated from a uniform distribution over a box aligned with the principal components. This corresponds with choice b of Tibshirani, Walther, and Hastie (2001, p. 414) who show that this option gives the best results. In detail, the GAP statistic is given by

$$\mathrm{GAP}(k) = \frac{1}{B}\sum_{b=1}^{B} \log(\mathrm{SSR}_k(b)) - \log(\mathrm{SSR}_k)$$

where SSR_k is the aggregated sum of the squared orthogonal distances for the original data set split into k groups. Similarly, $\mathrm{SSR}_k(b);\ b = 1, \ldots, B$ is the aggregated sum of the squared orthogonal distances for a data set generated from the reference distribution and split into k groups. Following Tibshirani, Walther, and Hastie (2001) we select the optimal number of groups $\hat{k}$ as follows

$$\hat{k} = \text{ smallest } k \text{ such that } \mathrm{GAP}(k) \geq \mathrm{GAP}(k+1) - s_{k+1}$$

where $s_{k+1} = \mathrm{sd}_{k+1}\sqrt{1 + 1/B}$ with sd_{k+1} the standard deviation of the $\mathrm{SSR}_{k+1}(b)$ values. An advantage of the GAP statistic is that it is also defined for $k = 1$ group and hence can indicate whether the data contains several groups or not.

Alternatively, we consider criteria based on the log-likelihood of the data with a penalty term for the number of parameters. The penalty term $\rho(m, n)$ can depend on the sample size n and the number of parameters in the model m. To compute the likelihood, we use the following model for the j^{th} group $(j = 1, 2, \ldots, k)$

$$x_i = \mu_j + A_j\gamma_i + \epsilon_i \qquad i = 1, \ldots, n_j \tag{1}$$

with $\epsilon_i \sim N(0, \sigma_j^2 I)$. Here A_j is a $d \times (d-1)$ dimensional orthogonal matrix and γ_i is a $d-1$ dimensional vector giving the scores of x_i in the $d-1$ dimensional hyperplanes. μ_j is the group center estimated by the sample mean $\bar{x}_j$. The matrix A_j is estimated as $\hat{A}_j = (a_1, \ldots, a_{d-1})$ where the a_i are the eigenvectors corresponding to the $d-1$ largest eigenvalues of the group covariance matrix. Finally, $\hat{\gamma}_i = \hat{A}_j^t(x_i - \bar{x}_j)$ and $\hat{\sigma}_j^2 = \frac{1}{n_j}\sum_{i=1}^{n_j} \|x_i - \hat{\mu}_j - \hat{A}_j\hat{\gamma}_i\|^2$. The corresponding log likelihood is given by

$$l_j(x_1, \ldots, x_{n_j}, \hat{\mu}_j, \hat{A}_j, \hat{\gamma}_i, \hat{\sigma}_j^2) = -\frac{n_j}{2}\log(2\pi) - \frac{n_j}{2} - \frac{n_j d}{2}\log(\hat{\sigma}_j^2).$$

Now combining the k groups and taking into account that the group sizes are unknown, we obtain the log likelihood

$$l(x_1, \ldots, x_n) = \sum_{j=1}^{k} n_j \log(n_j) - n\log n - \frac{n}{2}\log(2\pi) - \frac{n}{2} - \frac{d}{2}\sum_{j=1}^{k} n_j \log(\hat{\sigma}_j^2).$$

Note that the number of parameters in this model equals $m = k(d+1) + k(d-1)d/2 + n(d-1)$. Following Smith, and Spiegelhalter (1980) we used the following penalties: $\rho_1(m, n) = m\log(n)/2$ (BIC), $\rho_2(m, n) = m$ (AIC), $\rho_3(m, n) = 3m/4$ (local Bayes factor), $\rho_4(m, n) = m/2$, $\rho_5(m, n) = 3m/2$, $\rho_6(m, n) = 2m$ and $\rho_7(m, n) = m\log\log(n)/2$. For each of these methods we determine the optimal number of groups by selecting the value $\hat{k}$ for which the penalized likelihood is maximal. Since the penalized likelihoods are defined for $k = 1$, also these methods can indicate whether data contain several groups or not.

We conducted a simulation study to compare the GAP statistic and the seven penalty methods described above. We considered the following designs.

(1) 2-dimensional data of size $n = 300$ consisting of 3 separated groups of equal size in 2 dimensions. Figure 5a shows an example dataset and the LGA solution for $k = 3$.
(2) Normal data. The 5-dimensional data of size $n = 100$ are multivariate normal with covariance matrix $\Sigma = \text{diag}(5, 4, 3, 2, 1)$, hence $k = 1$.
(3) 2-dimensional data sets generated according to two crossing lines yielding overlapping groups of size 100. See Figure 5b for an example with the LGA solution for $k = 2$.
(4) 2-dimensional data containing two groups of size 100 that overlap at the left but are more separated at the right side as can been seen from Figure 5c.
(5) 2-dimensional data of size 50 with 2 groups, one close above the other, so the groups are not well separated. See Figure 5d for an example of the generated data with the LGA solution for $k = 2$.
(6) 2-dimensional data of size 75 with 3 groups closely on top of each other as shown in Figure 5e.

Note that for each of the simulation setups LGA applied with the correct number of groups detects the true groups corresponding to the data generating process as shown by the examples in Figure 5. This shows the capability of LGA to detect linear structures even when the different groups are heavily overlapping or not well separated.

The results of our simulation are reported in Table 2. The first row for each simulation setup gives the number of times the correct number of groups was selected by the method (out of 100). In the second row for each simulation setup we consider the wrong number of groups most frequently selected by the method. The wrong number of groups is shown in brackets and the value in the table is the number of times this wrong number was selected

From Table 2 we clearly see that the GAP statistic outperforms the likelihood based methods. Moreover, the GAP statistic can be expected to give the correct answer except in difficult situations with not well separated groups where only subject matter can give you guidance on the number of groups. The GAP statistic also behaves conservatively, that is, it never overestimates the number of groups contrary to the other methods that tend to overestimate the number of groups in some settings.

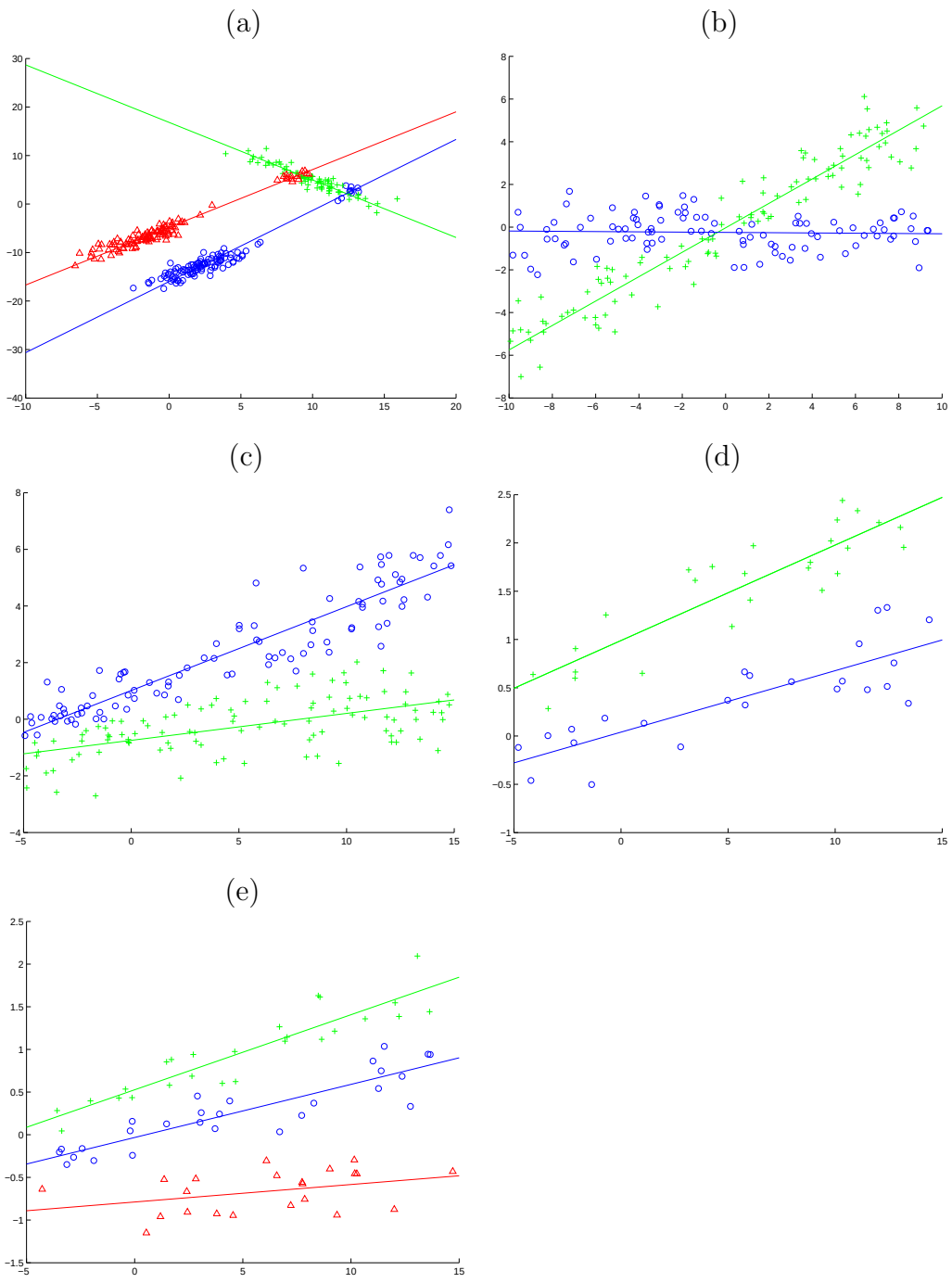

Fig. 5. Example data sets with LGA solution for the correct number of groups for the setup of simulation 1(a), 3(b), 4(c), 5(d), and 6(e).

4 Applications

In this section, we apply LGA to real problems in allometry and sports. These applications illustrate how LGA can be used as an exploratory tool in the analysis of real data.

Table 2
Simulation results to evaluate the performance of several criteria to estimate the optimal number of groups. The values in the table are the number of times (out of 100) the correct number of groups was selected (top line for each simulation) or the most frequently selected wrong number of groups (bottom line for each simulation) by the methods. The numbers between brackets are the most frequent selected wrong number of groups for the method.

	GAP	BIC	AIC	LBF				
		ρ_1	ρ_2	ρ_3	ρ_4	ρ_5	ρ_6	ρ_7
Simu 1 (k=3)	98	77	72	71	71	71	76	75
Simu 1	2(2)	12(4)	15(4)	14(4)	14(4)	15(4)	13(4)	13(4)
Simu 2 (k=1)	100	2	0	0	0	0	0	0
Simu 2	—	56(3)	91(4)	93(4)	95(4)	93(4)	61(4)	77(4)
Simu 3 (k=2)	89	0	0	0	0	0	0	0
Simu 3	11(1)	100(1)	95(1)	92(1)	82(1)	93(1)	99(1)	97(1)
Simu 4 (k=2)	39	0	3	2	2	3	1	3
Simu 4	61(1)	92(1)	48(1)	39(1)	44(3)	46(1)	80(1)	68(1)
Simu 5 (k=2)	97	2	6	10	15	10	2	4
Simu 5	3(1)	96(1)	66(1)	45(1)	44(4)	39(1)	96(1)	89(1)
Simu 6 (k=3)	11	0	6	3	3	3	0	1
Simu 6	72(1)	89(1)	38(1)	39(4)	55(4)	39(4)	84(1)	70(1)

4.1 Allometry data

In allometry studies biologists investigate the relationships between sizes of organs for different species. It often occurs in nature that if the size of one organ is large, then the size of other organs is also large because their biological functions are coordinated. For example, a larger body also requires a larger brain. These relations are driven by the evolution process. Typically, for certain species, say mammals, there exists a linear association between the (transformed) sizes, measured in weight or volume, of two organs. However, across different classes of species, the linear associations are not the same because of different living habits, environment, food sources, etc. Hence, grouping according to different linear patterns is necessary. In the past, assignments were done manually by biologists according to their scientific experience (see e.g. Jerison, 1973). This manual work of course is tedious and requires a lot of time to check each individual species. Here, we apply LGA to two allometry datasets to investigate whether LGA can match the results obtained by manual assignment.

In the first example the relationship between olfactory bulb volume and brain weight is investigated. Figure 6 shows the scatterplot of $\log_{10}$-olfactory bulb volume against $\log_{10}$-brain weight for 83 mammal species. (The data are courtesy of Prof. Jerison.) Roughly speaking, olfactory bulb volume increases with brain weight. However, also the variation of the log-olfactory bulb volume increases. For example, some species of monkeys have roughly the same brain weight as horses, but the latter has much larger olfactory bulb volume. During the evolution process, different mammal species have developed their smell senses according to their living environment, food searching and danger identifying needs, etc. Thus, the observed heteroscedasticity is due to the combination of different types of mammal species. Based on biological knowledge, Jerison (1973) divided the mammals into three groups: one including insectivores, carnivores and horses, one including prosimians (primitive primates characterized by nocturnal habits), and one including anthropoids (monkeys, apes, human). Then he fitted three separate regression lines to these groups, each exhibiting reasonable homoscedasticity. This suggests that there are three linear patterns among these species. Now we use LGA to see if it captures the three linear patterns. The result in Figure 6b shows that the majority of each group matches the division based on biological knowledge in the top panel.

From Figure 6a we see that the three groups are not well separated, so it will be difficult to detect the correct number of groups by the automatic procedures in the previous section. Indeed, all methods underestimate the number of groups in this data set and yield 2 groups as the optimal number. Hence, without using biological knowledge we would choose $k = 2$ groups as displayed in Figure 6c. This grouping is very reasonable. With few exceptions, the first group (labeled 1) includes insectivores, prosemians, carnivores and horses while the second (labeled 2) includes apes, monkeys, and humans. On the other hand, using the biological knowledge we would determine three groups. In this case, the grouping is very close to the manual biological solution with prosimians forming a separate group as discussed above.

The second allometry example studies the relationship between the brain and body weight for $n = 282$ vertebrates obtained from Crile, and Quiring (1940). The scatterplots of $\log_{10}$-brain weight against $\log_{10}$-body weight in Figure 7 resembles that in Jerison (1973, page 43), but with more species. The scatterplots in Figure 7 show an increasing relation between brain and body weight. However, a closer look reveals that there may be different linear groups. Jerison (1973) argued that this data consists of four groups that could as well be merged into two groups. The two main groups are the higher vertebrates consisting of birds and mammals (such as bat, crow, baboon, chimpanzee, lion, dolphin, elephant, whale, etc) and the lower vertebrates consisting of fish, reptiles and amphibians (like goldfish, eel, latimeria, alligator, etc). With $k = 2$, LGA gives the result in Figure 7(a). Although there are a few assignment errors in the top and right region, the majority of each group matches the

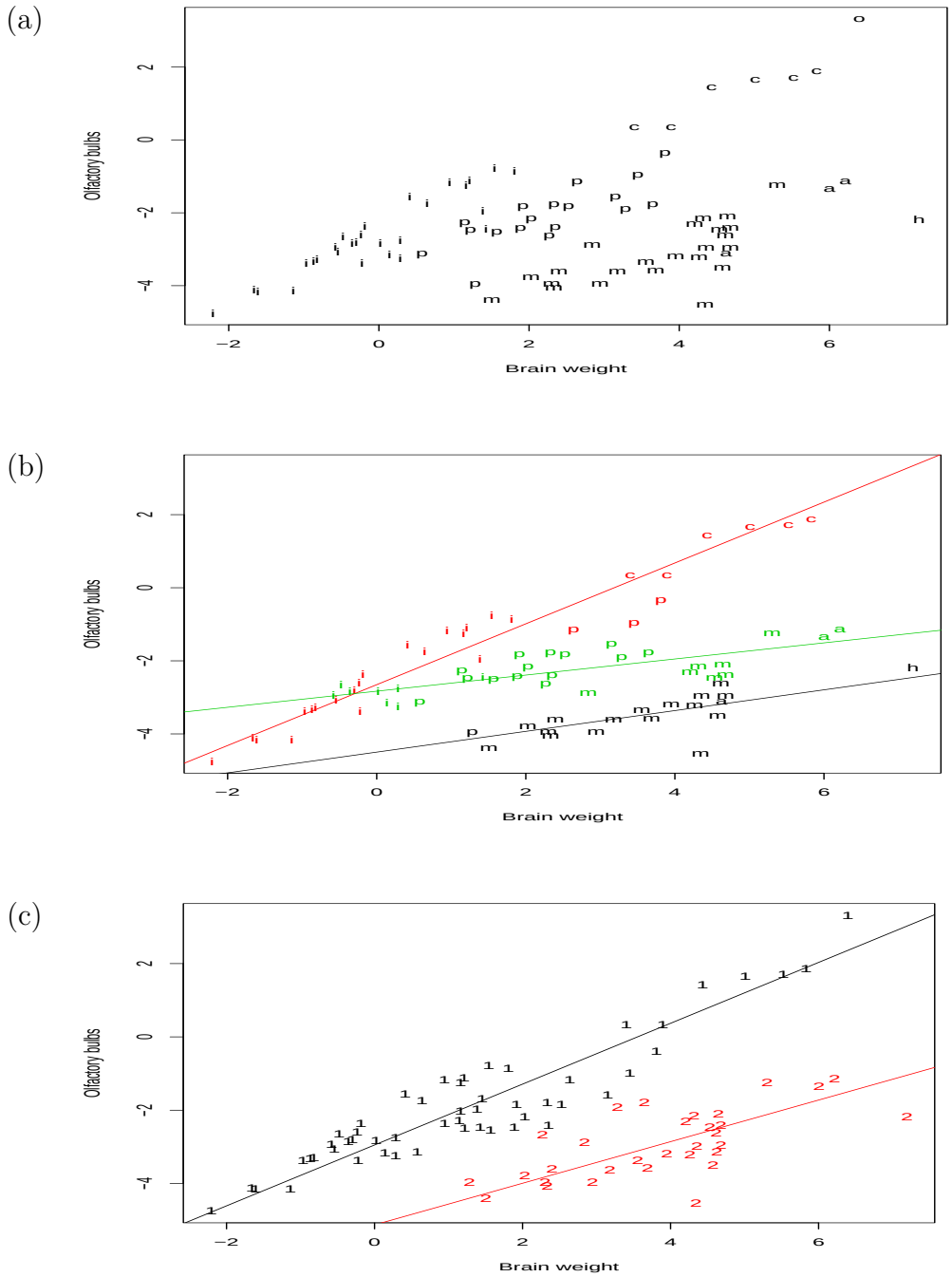

Fig. 6. (a): Logarithms of Olfactory bulbs vs. brain weight for some mammal species: insectivores (i), carnivores (c), prosimians (p), apes (a), monkeys (m), human (h) and Horse (o). (b): Three groups detected by LGA. (c): Two groups detected by LGA.

biological division.

Both the higher and lower vertebrates have been split further into finer subgroups. The higher vertebrates can be divided into a subgroup of birds and a subgroup of mammals (including primates), while the lower vertebrates can be separated into a subgroup of fish and a subgroup of reptiles. However, from the scatterplot we can not see these finer patterns because they highly overlap each other within the higher and lower vertebrates. Thus, based on graphical representation of the data without scientific knowledge we cannot detect these finer partitions. With $k = 4$, LGA gives the result in Figure 7(c) where as expected, each main group has been split into two subgroups. With very few exceptions the four groups match the existing biological division. This confirms that LGA is capable of revealing linear groups even if the linear patterns lie close together.

Finally, without biological knowledge we would need to rely on the automatic procedures to determine the 'appropriate' number of groups. The GAP statistic applied for these data yields $k = 1$. This conservative result is not surprising since the groups are not well separated. The likelihood based methods all give $k = 3$ as the optimal number of groups. This could be seen as an overestimation of the appropriate number of groups ($k = 2$), but also this grouping seems to make biological sense as shown in Figure 6(b): The higher vertebrates are separated (as in Figure 6(c)) and the lower vertebrates form the third group.

4.2 Hockey data

We now analyze a dataset containing information on the performance of players in the Canadian National Hockey League for the 94-95 competition. For each of the 871 players we consider 4 variables measured during the hockey season:

PTS:	Points scored (this is the total of goals and assists)
P/M:	+/- average rating, $+1(-1)$ if team(opponent) scored in an even-strength situation
PIM:	Total penalty time in minutes
PP:	Power play goals

These variables reflect the strength of both attackers and defenders. Our goal is to discover knowledge from this hockey-related dataset. Although some patterns might be obvious for a hockey expert, we will use LGA to identify potential groups among the players without using any knowledge of each player and his team.

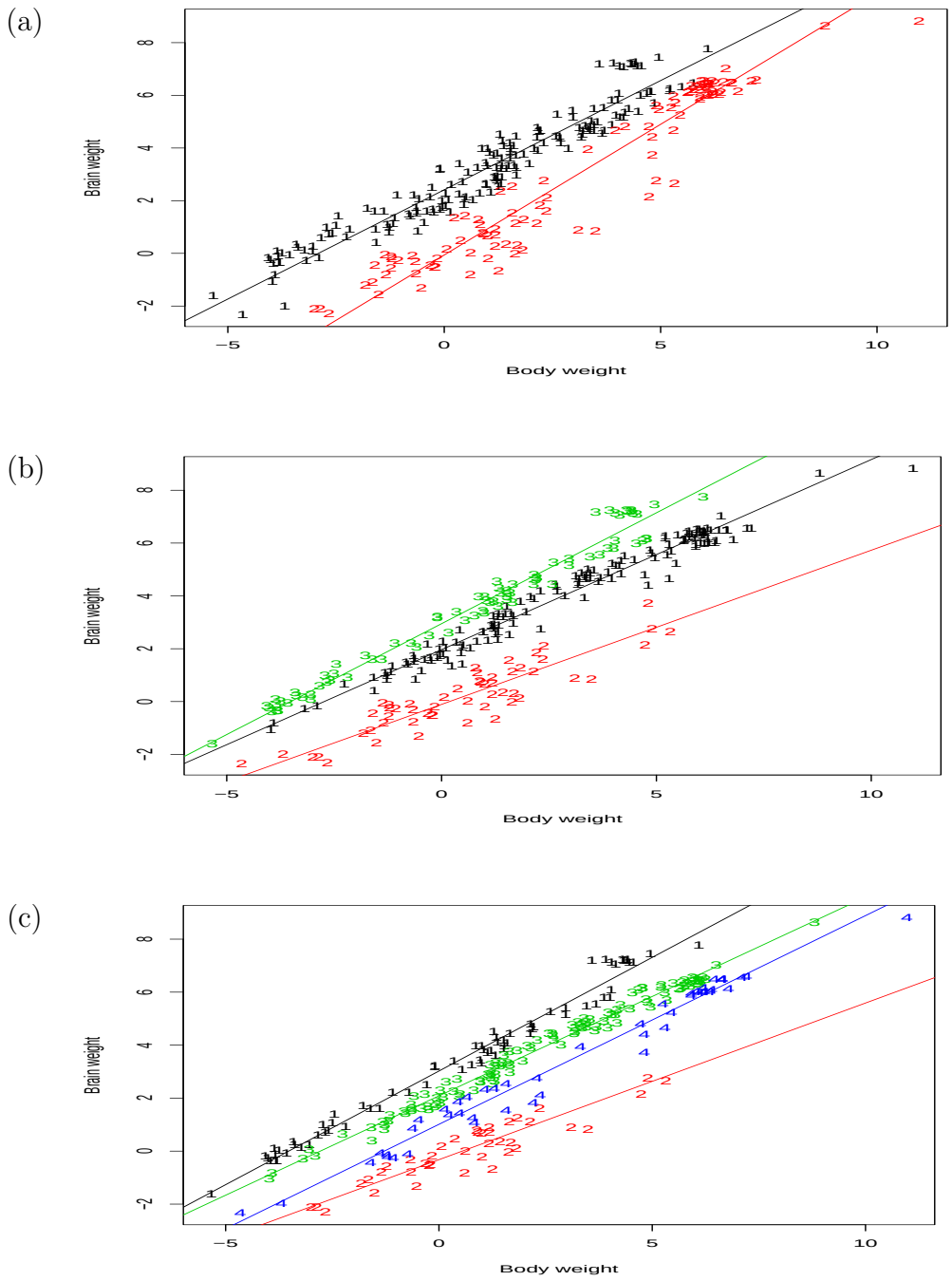

Fig. 7. Logarithms of brain weight vs. body weight for 282 vertebrate species for (a) two, (b) three, and (c) four groups. They are all detected by LGA.

Note that in this case there is no obvious response variable, but LGA doesn't require one. Moreover, we don't know which variables are useful to separate the players in different linear groups. Fortunately, LGA is capable of uncovering important grouping variables as will be discussed further in section 7. Since there is no previous knowledge about the number of groups, we apply our procedure to identify k. The GAP statistic behaves conservative and selects $k = 1$, meaning that there are no groups. On the other hand, the likelihood based methods all select $k = 3$. However, the latter solution may be overly optimistic, so let us investigate further the LGA solution for $k = 3$ to see whether this grouping makes sense.

The coefficients of the three hyperplanes are given in Table 3. We see that for all hyperplanes the coefficients of the variables P/M and PIM are very small compared to the coefficients of PTS and PP. Note that all variables are standardized by LGA so these coefficients can be compared directly. Thus, the two variables PTS and PP seem to be more useful for our purposes. This demonstrates the capability of LGA to identify informative variables as will be investigated further in section 7.

Table 3
Orthogonal regression coefficients of the 4 variables for the three groups.

Group	PTS	P/M	PIM	PP
1	-0.156	0.015	0.001	0.988
2	-0.221	0.029	-0.003	0.975
3	0.113	-0.010	0.001	-0.994

Figure 8 shows the scatterplot of all players divided into three groups. We can see that there are quite a number of points near the origin, which correspond to the defenders who are seldom active in attacking. As we move away from the origin, the lines summarizing the groups become more distinct. The lower group might represent the "team players" that can score and make assists but seldom play in power play situation. The upper group might represent the 'sharp shooters' who score many goals and often play in power play situations. Finally, the middle group are second choice shooters for power play situations. The statements are based on the average performance of the players through the season. The fact that a player belongs to the upper group does not necessarily mean that he should always be put on the ice when a power play is immediate. Since hockey is a team sport we should not draw any naive conclusions from the data such as who should be playing at what occasion. Any valid conclusion would require a detailed and more careful analysis. However, LGA does provide a very good starting point for such analysis.

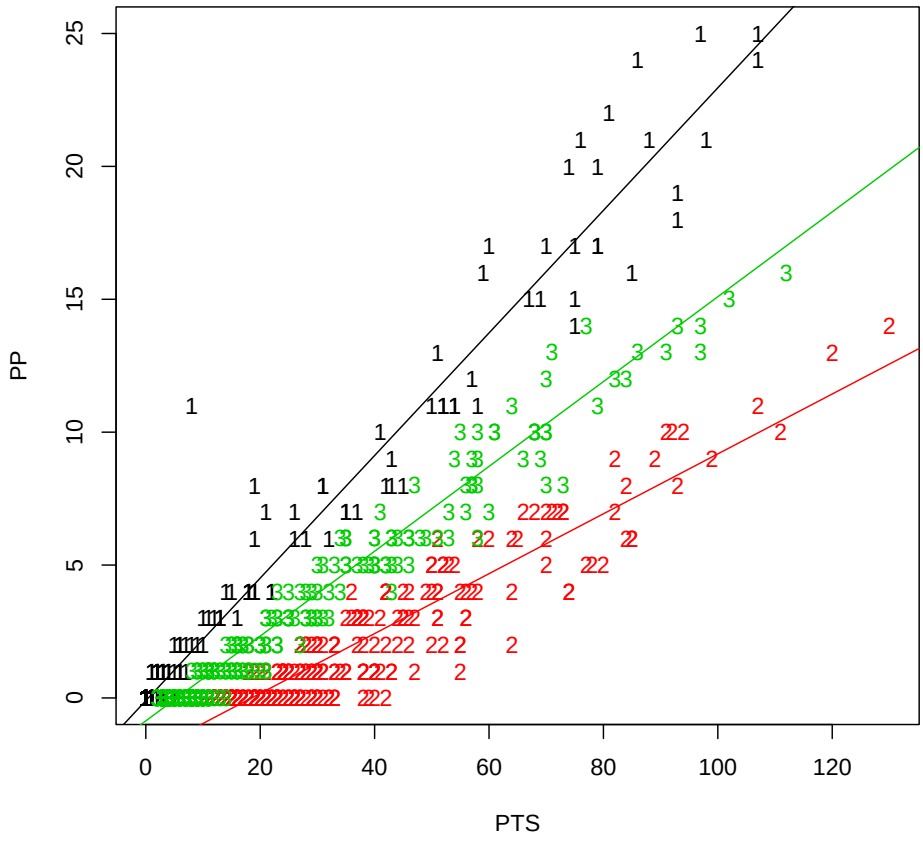

Fig. 8. Plot of PP versus PTS for the NHL 94-95 competition with the three groups detected by LGA.

5 Measuring strength of group membership

Wrongly assigned objects are inevitable with any grouping method. In the case of different linear groups, it is especially obvious that assignments are difficult in the intersection regions between two (or more) hyperplanes. Consider for example the intersections in Figure 9(a). Also when two (or more) groups heavily overlap (as in Figure 7(c)) errors will be made. Points in these 'intermediate' regions will be close to more than one hyperplane and could be given double or multiple membership.

For partitioning methods in clustering, Rousseeuw (1987) introduced the silhouette width of an object to measure how strongly an object belongs to the cluster it has been assigned to. We adapt the definition of silhouette width of an object for the case of linear groups. Recall that each group is characterized by a hyperplane and each object is assigned to the closest hyperplane. Apart

from the assigned group, for each object we can also define its neighbor which is the second closest hyperplane. The silhouette width of an object compares the distance to the assigned group with the distance to its neighbor. Denote $s(i, j)$, the squared distance between object i ($i = 1, 2, \ldots, n$) and hyperplane j ($j = 1, 2, \ldots, k$). Denote the two smallest values of $s(i, j)$ by $s_1(i)$ and $s_2(i)$, respectively. Then the silhouette width for object i is defined as

$$w(i) = 1 - \frac{s_1(i)}{s_2(i)}, \qquad i = 1, 2, \ldots, n. \tag{2}$$

Note that $0 \leq w(i) \leq 1$ because $0 \leq s_1(i) \leq s_2(i)$. If $s_1(i) = s_2(i)$ (that is, object i can equally well be assigned to its neighbor) then $w(i) = 1 - 1 = 0$, the lower bound. If $s_1(i)/s_2(i) \to 0$ (that is, object i is much closer to the assigned group than to its neighbor) then $w(i) \to 1$, the upper bound. Thus, the silhouette width measures how strongly each object belongs to its assigned group. The larger the silhouette width of an object, the more confident one can be about the correctness of its assignment. On the other hand, objects with smaller silhouette widths are more likely to be assigned incorrectly.

Suppose that LGA splits the dataset into k groups denoted by C_j with number of objects n_j ($j = 1, 2, \ldots, k$). Then the average silhouette width for Group j, given by

$$\bar{w}_j = \sum_{i \in C_j} w(i)/n_i, \qquad j = 1, 2, \ldots, k,$$

measures the strength of that group, that is, how well this group is separated from the other groups. A high average width means that a well defined group has been found while a low average width means that not much structure has been detected. Finally, the average silhouette width of all objects

$$\bar{w}(k) = \sum_{i=1}^{n} w(i)/n$$

measures the strength of the grouping when the number of groups equals k. A high overall average corresponds to a strong structure while a low average corresponds to a weak structure. Hence, the overall average silhouette width $\bar{w}(k)$ can be used as a diagnostic to evaluate whether an LGA solution yields a reasonable structure or not.

As an illustration we compute the silhouette widths for the slanted π synthetic dataset in Figure 9 (generated by random points from three linear models). The silhouette plot (Rousseeuw, 1987) for $k = 3$ groups in Figure 10(a) shows the silhouette widths for the points in each of the three groups (going from

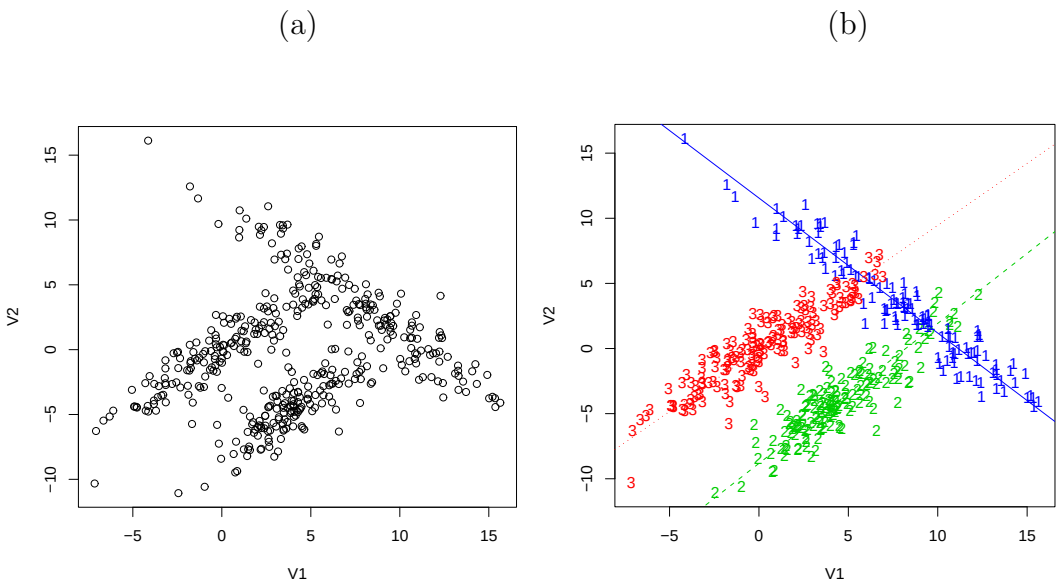

Fig. 9. (a) Slanted π data set (b): LGA solution for $k = 3$ groups.

smallest to largest silhouette value within each group). Most of the points have a silhouette value above 0.50 meaning that the distance to their neighbor is at least twice the distance to their group. This strong structure is confirmed by the group averages which are $\bar{w}_1 = 0.70$, $\bar{w}_2 = 0.82$, and $\bar{w}_3 = 0.82$, showing that the three groups are well separated. As expected, also the overall average $\bar{w}(3) = 0.78$ is high. Figure 10(b) shows the three groups and the points with silhouette width less than 0.25 (that is, the distance to their neighbor is at least 3/4 of the distance to their group). We clearly see that these points all lie in the intersection regions.

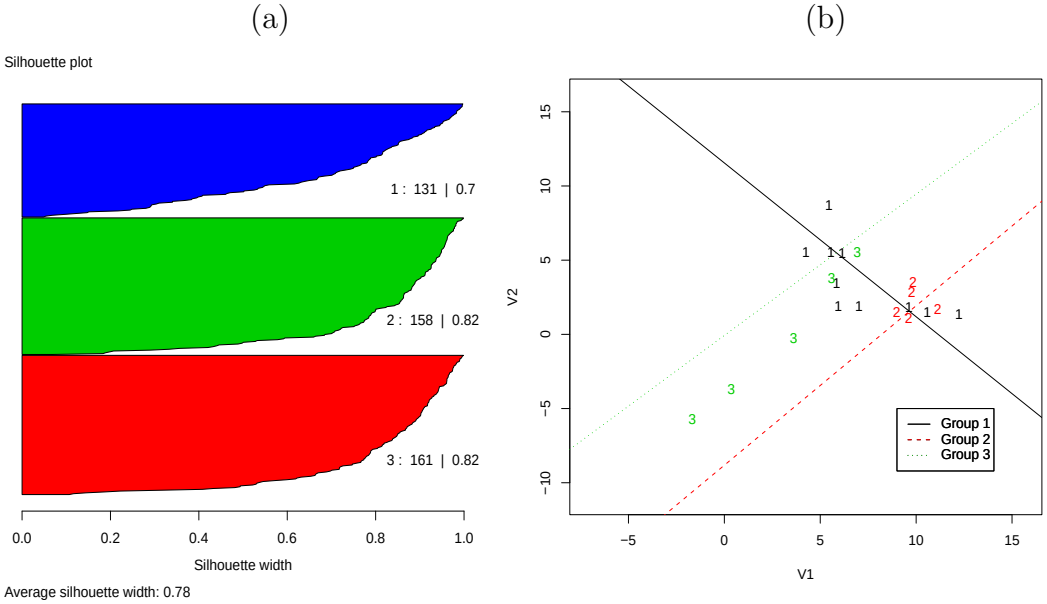

Fig. 10. (a): Silhouette plot of the linear grouping with $k = 3$ (b): Plotted objects have silhouette width less than 0.25.

Figure 11 shows the silhouette plot for the tree data with $k = 2$ groups as in Figure 4(b). The average group silhouette widths are $\bar{w}_1 = 0.78$ and $\bar{w}_2 = 0.85$. The groups are well separated since no points have silhouette width below 0.25 and all but three points have silhouette width above 0.5. This plot thus confirms that a strong structure is detected.

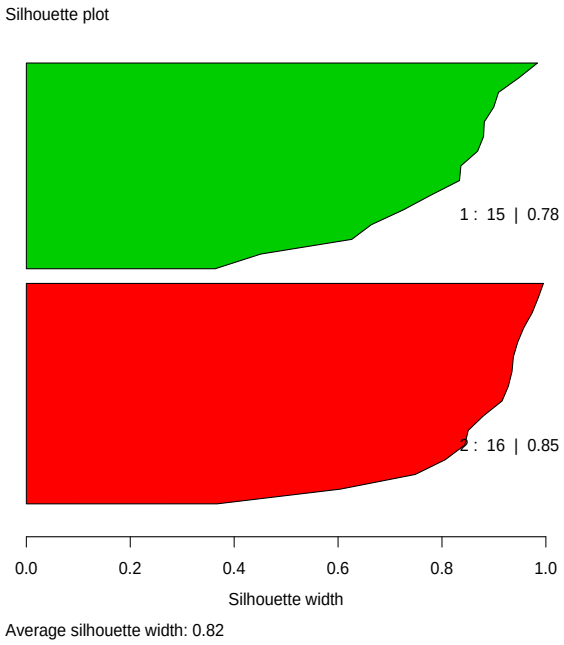

Fig. 11. example of Figure 4 continued. Silhouette plot of the linear grouping with $k = 2$.

For each group, the points with high silhouette width (e.g. ≥ 0.50) best represent the group. Comparing these points among and between groups can often help to find one or several common features that are on the one hand shared by the points in a group and on the other hand distinguish this group from the other groups in the dataset.

Silhouette values have a nice and easy interpretation in terms of distances from the respective hyperplanes but are computationally expensive for large data sets. As an alternative the strength of group membership can be determined using posterior distributions. For each observation x_i we can determine the Bayes factor based on the model (1)

$$\mathrm{BF}(i) = \log\left(\frac{\hat{\pi}_2\,\hat{f}_2(x_i)}{\hat{\pi}_1\,\hat{f}_1(x_i)}\right)$$

where $\hat{f}_1(x_i)$ is the density of x_i for the group it is assigned to, and $\hat{f}_2(x_i)$ is the density of x_i for its neighbor. The neighboring group of an observation x_i is determined as the group for which $\hat{\pi}_j\,\hat{f}_j(x_i)$ is maximal among all groups

not containing x_i. If x_i clearly belongs to its assigned group, then the denominator in the Bayes factor will be much larger than the numerator yielding a large negative value. On the other hand, if x_i is an intermediate point, then $\hat{\pi}_2 \hat{f}_2(x_i) \approx \hat{\pi}_1 \hat{f}_1(x_i)$ such that the Bayes factor is close to 0. Note that the Bayes factors take the group sizes into account. Similarly as for silhouette widths we can define the average Bayes factor for each group as

$$\overline{BF}_j = \sum_{i \in C_j} BF(i)/n_i, \qquad j = 1, 2, \ldots, k,$$

which measures the strength of each group. The overall average Bayes factor of all objects is given by

$$\overline{BF}(k) = \sum_{i=1}^{n} BF(i)/n.$$

and measures the strength of the grouping when using k groups.

Figure 12(a) shows the Bayes factor plot for the slanted π data set. Most points have a Bayes factor much smaller than $\log(0.5) = -0.69$ indicating that they clearly belong to their assigned group. The strong structure is also confirmed by the group averages of $\overline{BF}_1 = -8.35$, $\overline{BF}_2 = -16.99$, and $\overline{BF}_3 = -17.37$, and by the overall average of $\overline{BF}(3) = -14.61$. Figure 12(b) shows the points with Bayes factor larger than $\log(3/4)$. Again we clearly see that these points all lie in the intersection regions showing that large Bayes factors correspond to intermediate points.

6 Generalized LGA

As suggested by the Associate Editor, one can consider the more general problem of finding k groups of points around hyperplanes of dimensions $0 \leq l_i \leq d-1$ $(i = 1, 2, ..., k)$, with the case $l_i = 0$ corresponding to a group concentrated around a single point.

In general a $d - j$ dimensional hyperplane $(j \leq d)$ is given by the equation

$$Ax = B$$

where A is an orthogonal $j \times d$ matrix and B is a j-dimensional vector. Therefore we search for groups with "central hyperplanes" given by

$$(A_1, B_1), ..., (A_k, B_k),$$

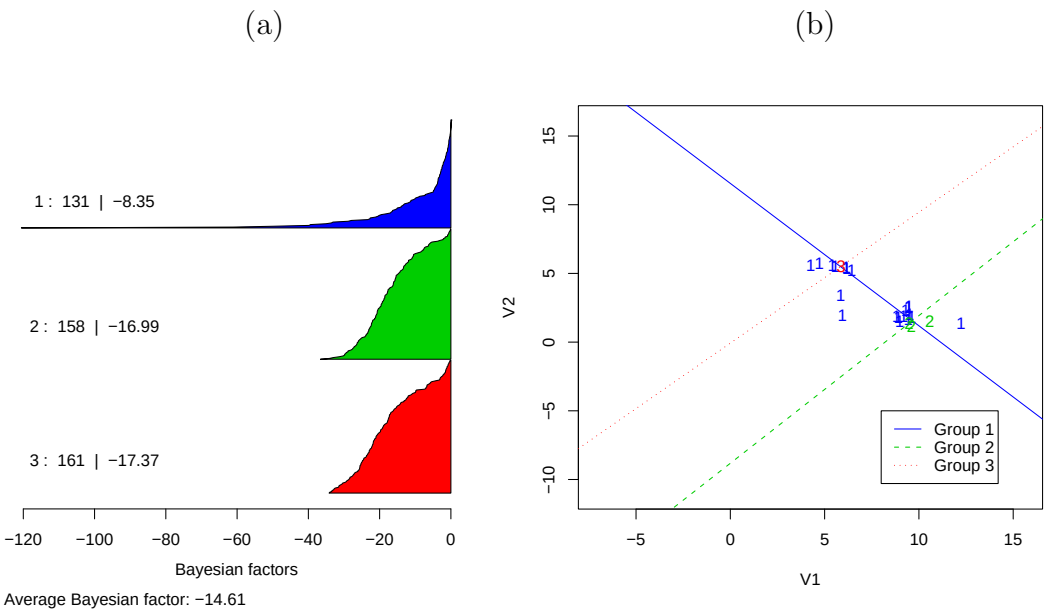

Fig. 12. (a): Bayes factor plot of the linear grouping with $k = 3$ (b): Plotted objects have Bayes factor larger than $\log(3/4)$.

where the dimension may vary from group to group.

We now describe a procedure to find $(A_1, B_1), ..., (A_k, B_k)$ using the basic LGA algorithm as a building block. Our procedure is then illustrated by a synthetic example.

Step 1. Finding homogenous groups.

(a) We start by applying LGA (equipped with an appropriate method to select the number of groups). For each of the detected groups we now have two possibilities. The group could be a homogeneous $d-1$ dimensional linear group or it may consist of one or more subgroups scattered in this $d-1$ dimensional hyperplane. To determine whether there is more than one group we apply LGA (again equipped with an appropriate method to select the number of groups) to the reduced dataset obtained by projecting the group points onto the corresponding $d-1$ dimensional hyperplane.

(b) We iteratively repeat the procedure in (a), with $d-1$ replaced by $d-j$, $j = 2, 3, ..., d-1$, to each new subgroup detected in the previous step. For example, we would apply LGA to the points in a new subgroup found in the $d-1$ dimensional hyperplane once they have been projected onto the corresponding $d-2$ dimensional hyperplane, and so on.

Step 2. Finding the appropriate dimension of each group. This procedure is applied to each homogeneous group characterized by a $d-j$ hyperplane from Step 1.

(a) To determine the dimension of each group we reason as follows. If a

group is well characterized by a hyperplane of dimension $d-j$, then the spread in the directions of the hyperplane are considerably larger than that of the residual spread perpendicular to the hyperplane. On the other hand, if the group can be characterized by a hyperplane of lower dimension, then there will be directions in the hyperplane in which the spread is comparable to the spread of the residuals. Spread can be measured by the eigenvalues of the covariance matrix of the group. Therefore, we wish to test the hypothesis $H_0 : \lambda_{d-j+1} = \lambda_{d-j}$. The corresponding likelihood ratio test statistic is given by

$$L_n\left(d-j\right) = 2\left(\hat{\lambda}_{d-j}\hat{\lambda}_{d-(j-1)}\right)^{\frac{n}{2}} / \left(\hat{\lambda}_{d-j} + \hat{\lambda}_{d-j+1}\right)^{n} \qquad (3)$$

Under H_0, $-2\log\left(L_n\left(d-j\right)\right)$ is asymptotically distributed as a χ^2_2 distribution (see e.g. Tyler, 1982). If the test rejects H_0, then λ_{d-j} is substantially larger than λ_{d-j+1} and we conclude that the current dimension $d-j$ is appropriate. That is, $A = \left(\mathbf{a}_d^{'}\right)$ when $j = 1$, and

$$A = \begin{pmatrix} \mathbf{a}_d^{'} \\ \mathbf{a}_{d-1}^{'} \\ \vdots \\ \mathbf{a}_{d-j+1}^{'} \end{pmatrix}, \quad \text{when} \quad j = 2, \ldots, d-1.$$

In general, $\mathbf{a}_i$ is the eigenvector associated with $\hat{\lambda}_i$. Otherwise, we reduce the dimension of the hyperplane by one, by adding one row to A: $A = \left(\mathbf{a}_d^{'}\right) \rightarrow A = \begin{pmatrix} \mathbf{a}_d^{'} \\ \mathbf{a}_{d-1}^{'} \end{pmatrix}$ in the case $j = 1$, and

$$A = \begin{pmatrix} \mathbf{a}_d^{'} \\ \mathbf{a}_{d-1}^{'} \\ \vdots \\ \mathbf{a}_{d-j+1}^{'} \end{pmatrix} \rightarrow A = \begin{pmatrix} \mathbf{a}_d^{'} \\ \mathbf{a}_{d-1}^{'} \\ \vdots \\ \mathbf{a}_{d-j}^{'} \end{pmatrix}, \quad \text{when} \quad j = 2, \ldots, d-1$$

(b) We iteratively pursue further possible dimension reductions by applying (3) to $\hat{\lambda}_{d-m}$ and $\hat{\lambda}_{d-m+1}$ $(m = j+1, \ldots, d-1)$ and continue to add an extra row to A, until $L_n\left(d-m\right)$ becomes significant.

To illustrate this procedure we generated the example data set shown in Figure 13. In the first step we applied LGA combined with the GAP statistic to the full data set. We obtained $\hat{k} = 2$ as optimal number of groups and the LGA solution consisted of a plane formed by the top group (marked $\triangle$) and a plane

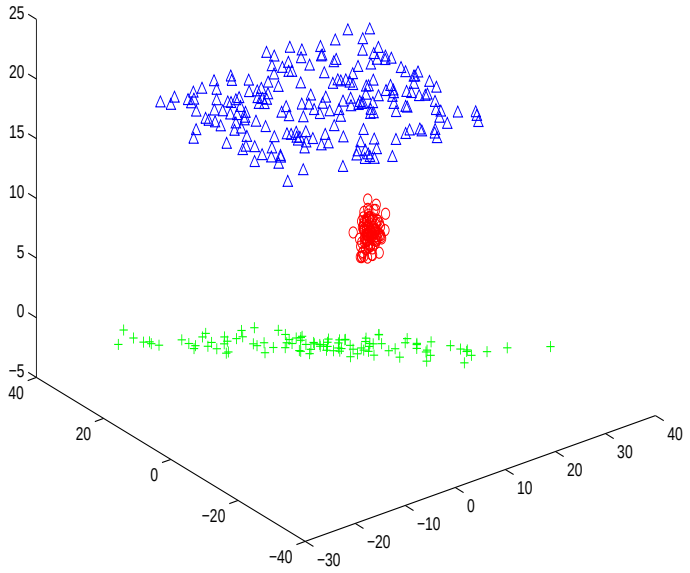

Fig. 13. Three-dimensional data set with 3 groups. One group (△) is concentrated around a plane, one group (+) is concentrated around a line and one group (o) is concentrated around a point.

formed by the 2 other groups. We projected both groups onto their respective central hyperplanes and applied LGA (with the GAP statistic again) to the projected datasets. For the top group we found $\hat{k} = 1$ so no further splits were necessary. For the bottom group we found $\hat{k} = 2$ and LGA split the group into a subgroup containing the points around the line (marked +) and a subgroup consisting of the points marked (o). No further splits were necessary after we found the given three homogeneous groups. Then we proceeded to the second step and used the likelihood ratio test to compare eigenvalues. For the top group the p-value was zero, indicating that this group is indeed a 2-dimensional linear group. For the bottom group, the p-value was 0.38 when comparing λ_2 with λ_3, indicating that the spread in these two directions is not significantly different. Therefore, we reduced the dimension of this group by 1. To determine whether this group is concentrated around a line (dimension 1) or around a point (dimension 0) we compared λ_1 with λ_2 which yielded a p-value of zero, leading to the conclusion that the bottom group is concentrated around a line. Finally, we compared the eigenvalues for the middle group. Comparing λ_2 with λ_3 gave a p-value equal to 0.164, so we reduced the dimension by 1. Comparing λ_1 with λ_2 gave a p-value of 0.85, so we again reduced the dimension by 1 and concluded that this group is concentrated around a single point.

7 Discussion

Clustering focuses on finding groups in data that are concentrated around different centers. We extend clustering to find groups in data that follow different linear relationships. Contrary to most of the existing literature, LGA aims at detecting functional linear relationships by using orthogonal regression. Hence, LGA has the advantage that a response variable is not needed. Moreover, we have illustrated that LGA also works well in the presence of nuisance variables that do not contribute to the linear grouping. The strengths of LGA makes it a useful tool in statistics and exploratory data analysis for finding interesting linear patterns.

The Hockey data set in Section 4 illustrates the capability of LGA to reveal linear patterns in the presence of nuisance variables. To further investigate this property of LGA we performed a small simulation study. We generated two-dimensional data sets of size 100 consisting of two equally sized groups. The first group concentrates around the line $5X_1 - X_2 = 0$ while the second group lies around the line $X_1 - X_2 = 50$. We generated X_1 according to $N(0, 100)$ while the errors come from $N(0, 25)$. For each of 100 such data sets we then added noise variables according to $N(0, 100)$. The number m of noise variables varies from 0 to 5. Table 4 shows the average absolute values of the coefficients (averaged over the 100 data sets) of the hyperplanes estimated by LGA with $k = 2$. The top half shows the results for the group with pattern $5X_1 - X_2 = 0$. Note that the corresponding standardized equation of the line is given by $0.98\,X_1 - 0.20\,X_2 = 0$. The bottom half corresponds to the pattern $X_1 - X_2 = 50$ whose standardized version is given by $0.71\,X_1 - 0.71\,X_2 = 35.36$. Comparing with the standardized coefficients above, we see from Table 4 that LGA captures the linear patterns well in the absence of noise variables. When a few noise variables are added ($m \leq 3$), LGA still captures the same patterns and the noise variables are easily identified by the small values of their coefficients compared to those of the important variables. When more noise variables ($m \geq 4$) are added, the behavior of LGA becomes less stable which indicates that LGA now misses the patterns in some of the data sets due to the added variability in the data.

We conducted a small simulation study to give an indication of the computation time needed by LGA. For dimension $d = 2, 3, 5$, and 10 and number of groups k going from 2 to 4 we generated 100 data sets with 25 points in each group and measured the average computation time needed by LGA applied with the true number of groups. The number of random starts for LGA was chosen according to Table 1 which assures that we have a clean start with 95% probability. The resulting computation times in seconds are shown in Table 5. These computation times were measured on a 1GHz Pentium using a MATLAB implementation of the LGA algorithm. Comparing these results

Table 4
Coefficients (in absolute value) of two hyperplanes estimated by LGA with $k = 2$ in the presence of noise variables. The top half shows the coefficients for the group concentrated around $5x_1 - x_2 = 0$ while the bottom half shows the results for the group around $x_1 - x_2 = 50$.

m	Constant	X_1	X_2	X_3	X_4	X_5	X_6	X_7
0	0.18	0.98	0.19					
1	0.19	0.98	0.19	0.018				
2	0.21	0.98	0.19	0.018	0.018			
3	0.32	0.98	0.19	0.025	0.033	0.025		
4	0.35	0.97	0.19	0.028	0.039	0.037	0.031	
5	0.54	0.95	0.18	0.042	0.043	0.036	0.048	0.051
0	32.69	0.64	0.76					
1	32.38	0.64	0.77	0.057				
2	32.25	0.63	0.77	0.058	0.053			
3	29.95	0.58	0.77	0.083	0.077	0.080		
4	26.56	0.51	0.75	0.122	0.096	0.104	0.111	
5	25.02	0.49	0.70	0.123	0.114	0.122	0.132	0.124

Table 5
Average computation times (in seconds) needed by LGA.

	k		
d	2	3	4
2	0.24	0.96	4.74
3	0.29	1.73	8.81
5	0.42	2.97	19.30
10	0.69	6.87	62.03

with Table 1 we see that larger increases in computation time correspond to increases in the required number of random starts.

The LGA algorithm proposed in this paper is not directly applicable to large data sets in high dimensions because similarly to K-means it does not scale well with the dimension. However, several improvements to K-means for data mining applications have been proposed (Bradley, Fayyad, and Reina, 1998). In future work we will investigate how LGA can be adapted for data mining applications.

Heteroscedasticity in a data set can be caused by the presence of more than

one linear structure close together. Such linear structures can be identified by LGA. However, heteroscedasticity can have several other causes. In such cases, LGA combined with one of the selection criteria for the number of groups can be overly optimistic. A number of groups exceeding one may be selected which leads to identification of spurious groups. If the cause of heteroscedasticity is not clear, we suggest to apply LGA with the GAP statistic which has the most conservative behavior and thus is least likely to overestimate the number of groups.

Another problem is handling outliers. It may occur that some part of the data does not follow any of the structures. Such data points could then be considered to be outliers for the method. However, like classical linear regression, orthogonal regression is very sensitive to outliers. This problem can be solved by using a robust orthogonal regression method (Zamar, 1989). In future work we will consider several robust proposals for orthogonal regression combined with robust clustering approaches (see e.g. Hardin and Rocke, 2004) to determine how the problem can be solved most efficiently.

8 Acknowledgment

We thank Prof. Harry Jerison at UCLA for his help in obtaining data sets, references and explanations for the allometry examples. We also thank David Zamar for his programming assistance. This research has been supported by a MITACS grant.

REFERENCES

References

Banfield, J.D. and Raftery, A.E., 1993. Model-based Gaussian and non-Gaussian clustering. Biometrics, 49, 803-821.

Bradley, P.S., Fayyad, U.M., and Reina, C.A., 1998. Scaling clustering algorithms to large databases. In: Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, AAAI Press, 9-15.

Crile, G. and Quiring, D.P., 1940. A record of the body weight and certain organ and gland weights of 3690 animals. The Ohio Journal of Science, XL, 219-259.

DeSarbo, W.S. and Cron, W.L., 1988. A maximum likelihood methodology for clusterwise linear regression. J. Classification, 5, 249-282.

DeSarbo, W.S., and Oliver, R.L., and Rangaswamy, A., 1989. A simulated

annealing methodology for clusterwise linear regression. Psychometrika, 54, 707-736.
Duda R.O. and Hart, P.E., 1973. Pattern classification and scene analysis. Wiley, New York.
Fisher, D.H., 1987. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2, 139-172.
Fukunaga, K., 1990. Introduction to statistical pattern recognition. Academic Press, San Diego, CA.
Fuller, W. A., 1987. Measurement error models. John Wiley and Sons, New York.
Gawrysiak, P., Okoniewski, M., and Rybiński, H., Clustering using regression as a mean of determining class quality. (Warsaw University of Technology, 2000).
Hardin, J. and Rocke, D.M., 2004. Outlier detection in the multiple cluster setting using the minimum covariance determinant estimator. Comput. Statist. Data Anal., 44, 625-638.
Hartigan, J. A., 1975. Clustering algorithms. Wiley, New York.
Hartigan, J. A., and Wong, M. A., 1979. A k-means clustering algorithm. Applied Statistics, 28, 100-108.
Jerison, H.J., 1973. Evolution of the brain and intelligence. Academic Press, New York.
Johnson, R.A. and Wichern, D.W., 1998. Applied Multivariate Statistical Analysis (Fourth Edition). Prentice Hall, New Jersey.
Kamgar-Parsi B., Kamgar-Parsi B., and Wechsler, 1990. Simultaneous fitting of several planes to point sets using neural networks. Computer Vision, Graphics and Image Processing, 52, 341-359.
Kaufman L. and Rousseeuw P.J., 1990. Finding groups in data. Wiley, New York.
Murtagh, F., 1983. A survey of recent advances in hierarchical clustering algorithms. The Computer Journal, 26, 354-359.
Murtagh, F. and Raftery, A.E., 1984. Fitting straight lines to point patterns. Pattern Recognition, 17, 479-483.
Murtagh, F., 2002. Clustering in massive data sets. In: J. Abello, P.M. Pardalos and M.G.C. Resende (Eds.), Handbook of Massive Data Sets, Kluwer, 401-545.
Ng, R.T., and Han, J., 1994. Efficient and effective clustering methods for spatial data mining. In: Proceedings of the 20th Conference on Very Large Databases, 144-155.
Pacheco, J. and Valencia, O., 2003. Design of hybrids for the minimum sum-of-squares clustering problem. Comput. Statist. Data Anal., 43, 235-248.
Phillips, T.-Y., and Rosenfeld, A., 1988. An ISODATA algorithm for straight line fitting. Pattern Recognition Letters, 7, 291-297.
Rousseeuw, P.J., 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math., 20, 53-65.
Rousseeuw, P.J. and Van Driessen, K., 1999. A fast algorithm for the minimum

covariance determinant estimator. Technometrics, 41, 212-223.
Ryan, T.A., Joiner, B.L., and Ryan, B.F., 1976. The Minitab student handbook. Duxbury Press.
Scott, D.W., 1992. Multivariate density estimation. Wiley, New York.
Silverman, B.W., 1986. Density estimation for statistics and data analysis. Chapman and Hall, London.
Smith, A.F.M., and Spiegelhalter D.J., 1980. Bayes factors and choice criteria for linear models. J. R. Stat. Soc. Ser. B Stat. Methodol., 42, 213-220.
Späth, H., 1982. A fast algorithm for clusterwise linear regression. Computing, 29, 175-181.
Späth, H., 1985. Cluster dissection and analysis. Ellis Horwood.
Tibshirani, R., Walther, G., and Hastie, T., 2001. Estimating the number of clusters in a data set via the gap statistic. J. R. Stat. Soc. Ser. B Stat. Methodol., 63, 411-423.
Tyler, D.E., 1982. Radial estimates and the test for sphericity. Biometrika, 69, 429-436.
Wedel, M. and Kistemaker, C., 1989. Consumer benefit segmentation using clusterwise linear regression. International Journal of Research in Marketing, 6, 45-59.
Woodruff, D.L. and Reiners, T., 2004. Experiments with, and on, algorithms for maximum likelihood clustering. Comput. Statist. Data Anal., 47, 237-253.
Zamar, R.H., 1989. Robust estimation in the errors in variables model. Biometrika, 76, 149-60.
Zhang, T., Ramakrishnan, R., and Livny, M., 1997. BIRCH: A new data clustering algorithm and its applications. Data Min. Knowl. Discov., 1, 141-182.